Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1.   (Currently Amended)   A method for optimizing the use of a plurality of processors when compiling a program in a computer system, the method comprising the steps of:

    (a)   ~~providing~~ locating a list of directories of the program and a list of processors of the computer system;

    (b)   assigning a next directory to a next available processor in an ordered manner to allow the next available processor to compile at least one file within the directory; ~~and~~

    (c)   compiling by the next available processor the at least one file within the next directory; and

    ~~(c)~~ (d)   repeating step (b) and (c) ~~to ensure that the maximum number of~~ until there are no more directories ~~can~~ to be compiled.

2.   (Original)   The method of claim 1 wherein the assigning step (b) further includes the step of (b1) obtaining a directory in which all dependencies have been satisfied.

3.   (Original)   The method of claim 1 wherein the assigning step (b) further includes the step of (b1), updating the list of processors and the list of directories based upon the assignment of the directory.

4.   (Original)   The method of claim 1 wherein the assigning step (b) further includes the step of (b1) providing a directory update mechanism for assigning the directories in the ordered manner.

5.   (Original)   The method of claim 4 wherein the providing an update mechanism step (b1) further comprises the steps of:

(b11) providing an array of dependency changes; and

(b12) merging the dependency changes array with a master array of changes.

6.   (Original)   The method of claim 5 wherein the merging step (b12) comprises the steps of:

(b121) obtaining a dependency change from the dependency changes array;

(b122) determining whether the dependency change is in a directory in the master array;

(b123) updating the directory in the master array of the dependency change in a directory of the master array;

(b124) adding dependency change to the master array in a new directory if the dependency change is not in a directory of the master array;

(b125) determining if there is another dependency change in the dependency changes array after either step (b123) or step (b124); and

(b126) repeating steps (b121) – (b125) until all dependency changes have been obtained from the dependency change array.

7. (Currently Amended) A system for optimizing the use of a plurality of processors when compiling a program in a computer system, the system comprising:

means for ~~providing~~ locating a list of directories in the program and a list of processors of the computer system;

means for assigning a next directory to a next available processor in an ordered manner to allow the next available processor to compile at least one file within the directory; ~~and~~

means for compiling by the next available at least one file within the next directory; and

means for ensuring that ~~the maximum number of~~ there are no more directories ~~can~~ to be compiled by utilizing the assigning means and the compiling means.

8. (Original) The system of claim 7 wherein the assigning means further includes means for obtaining a directory in which all dependencies and prerequisites have been satisfied.

9. (Original) The system of claim 7 wherein the assigning means further includes the means for updating the list of processors and the list of directories based upon the assignment of the directory.

10. (Currently Amended) The system of claim ~~10~~ 7 wherein the assigning means further includes the means for providing a directory update mechanism for assigning the directories in the ordered manner.

11. (Original) The system of claim 10 wherein the providing an update mechanism means further comprises:

means for providing an array of dependency changes; and

means for merging the dependency changes array with a master array of changes.

12.   (Currently Amended) ~~The method of claim 5~~ The system of claim 11 wherein the merging means comprises:

means for obtaining a dependency change from the dependency changes array;

means for determining whether the dependency change is in a directory in the master array;

means for updating the directory in the master array of the dependency change in a directory of the master array;

means for adding dependency change to the master array in a new directory if the dependency change is not in a directory of the master array;

means for determining if there is another dependency change in the dependency changes array; and

means for ensuring that all dependency changes have been obtained from the dependency change array.

13.   (Currently Amended) A computer readable medium containing program instructions for optimizing the use of a plurality of processors when compiling a program in a computer system, the program instructions for:

(a)   ~~providing~~ locating a list of directories of the program and a list of processors of the computer system;

(b)   assigning a next directory to a next available processor in an ordered manner to allow the next available processor to compile at least one file within the directory; ~~and~~

5

(c)   compiling by the next available processor the at least one file within the next directory; and

(e) (d)   repeating step (b) and (c) to ensure that the maximum number of until there are no more directories can to be compiled.

14.   (Original)   The computer readable medium of claim 13 wherein the assigning step (b) further includes the step of (b1) obtaining a directory in which all dependencies and prerequisites have been satisfied.

15.   (Currently Amended) The computer readable medium of claim 131 13 wherein the assigning step (b) further includes the step of (b1), updating the list of processors and the list of directories based upon the assignment of the directory.

16.   (Original)   The computer readable medium of claim 13 wherein the assigning step (b) further includes the step of (b1) providing a directory update mechanism for assigning the directories in the ordered manner.

17.   (Original)   The computer readable medium of claim 16 wherein the providing an update mechanism step (b1) further comprises the steps of:

(b11) providing an array of dependency changes; and

(b12) merging the dependency changes array with a master array of changes.

18.   (Original)   The computer readable medium of claim 17 wherein the merging step (b12) comprises the steps of:

(b121) obtaining a dependency change from the dependency changes array;

(b122) determining whether the dependency change is in a directory in the master array;

(b123) updating the directory in the master array of the dependency change in a directory of the master array;

(b124) adding dependency change to the master array in a new directory if the dependency change is not in a directory of the master array;

(b125) determining if there is another dependency change in the dependency changes array after either step (b123) or step (b124); and

(b126) repeating steps (b121) – (b125) until all dependency changes have been obtained from the dependency change array.